Towards Quality Assurance of Deep Learning Systems

Xie Xiaofei
Singapore Management University

14 December 2022

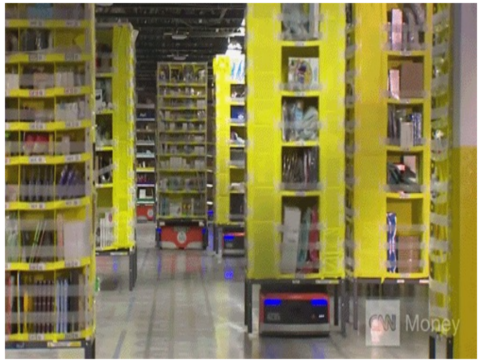
AI and Deep Learning are Revolutionizing our Society

Autonomous Driving

Manufacturing

Smart Devices

Logistics

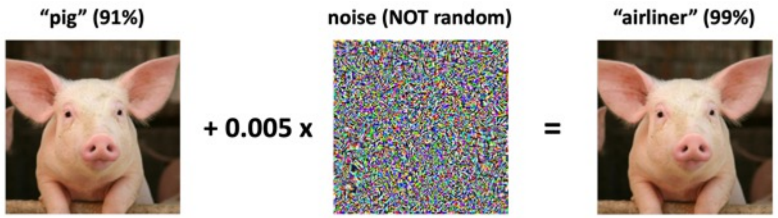


Malware Detection

Deep Learning Systems are Error-prone and Vulnerable

Weaknesses

Targeted Adversarial Inputs



Manual Object Tampering

Consequences


BBC Sign in News Sport Weather Shop Reel Travel

NEWS

Home | Video | World | Asia | UK | Business | Tech | Science | Stories | Entertainmen

Tesla in fatal California crash was on Autopilot

31 March 2018

Share

REUTERS

The driver of the Tesla Model X died shortly after the crash

➡ Quality assurance is in urgent need

* [Szegedy Zaremba Sutskever Bruna Erhan Goodfellow Fergus 2013], [Biggio Corona Maiorca Nelson Srndic Laskov Giacinto Roli 2013]

BBC Sign in News Sport Weather Shop Reel Travel

NEWS

Home Video World Asia UK Business Tech Science Stories Entertainment

Technology

AI image recognition fooled by single pixel change

3 November 2017

Share

BBC Sign in News Sport Weather Shop Reel Travel

NEWS

Home Video World Asia UK Business Tech Science Stories Entertainment

Technology

Psychedelic toasters fool recognition tech

3 January 2018

The New York Times

Alexa and Siri Can Hear This Hidden Command. You Can't.

Researchers can now send secret audio instructions undetectab… the human ear to Apple's Siri, Amazon's Alexa and Google's Assistant.

BBC Sign in News Sport Weather Shop Reel Travel

NEWS

Home Video World Asia UK Business Tech Science Stories Entertainment

Tesla in fatal California crash was on Autopilot

31 March 2018

Share

REUTERS

The driver of the Tesla Model X died shortly after the crash

AI to dominate banking, says report

28 March 2017

Share

GETTY IMAGES

Artificial intelligence will be the main way that banks interact with their customers within the next three years, a report from consultancy Accenture has suggested.

Banks such as Royal Bank of Scotland (RBS) are increasingly using chatbots to answer customer queries.

The report examined the views of 600 bankers and other experts.

Many, perhaps ironically, felt that AI would help banks create a more human-like

日本経済新聞

2018年11月20日（火）

人間は自動運転車を信頼できる？

自動運転 BP速報

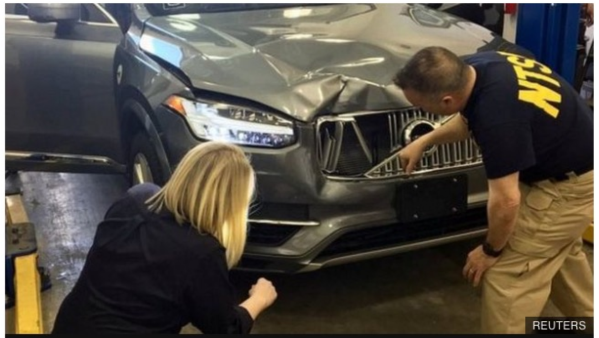
2018/8/31 23:00

Technology

Uber car 'had six seconds to respond' in fatal crash

24 May 2018

Share



REUTERS

National Transportation Safety Board investigators have examined the vehicle involved in the crash

BY RAY GRONBERG

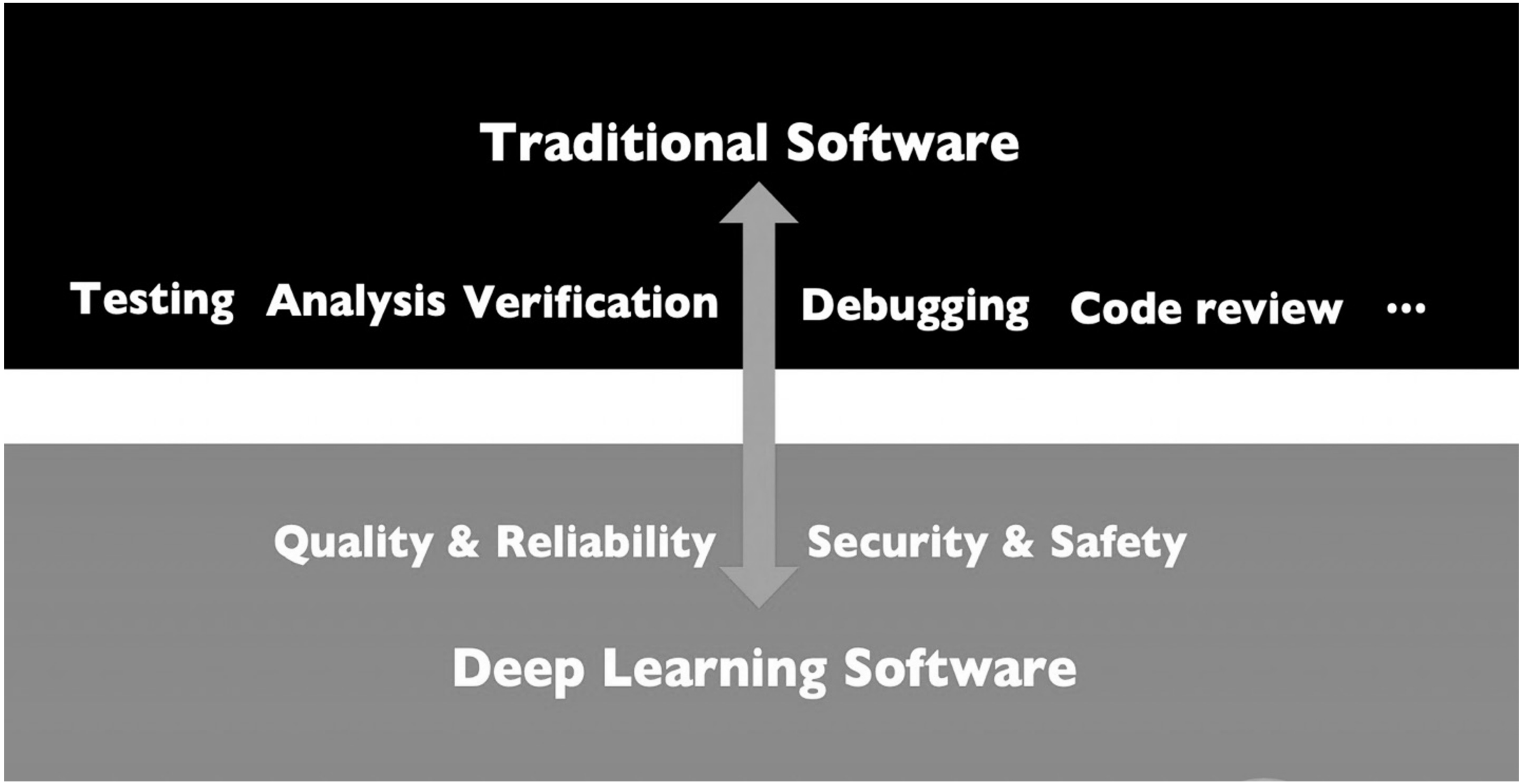
rgronberg@heraldsun.com

June 01, 2018 07:14 PM

Updated June 01, 2018 07:39 PM

Traditional Software and DL Software

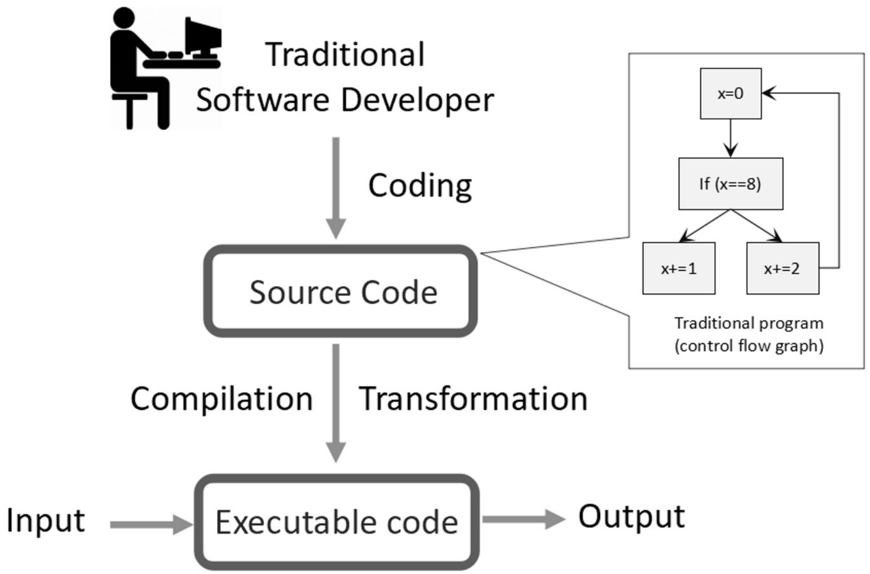
From Traditional Software to Deep Learning



Traditional Software vs Deep Learning Software

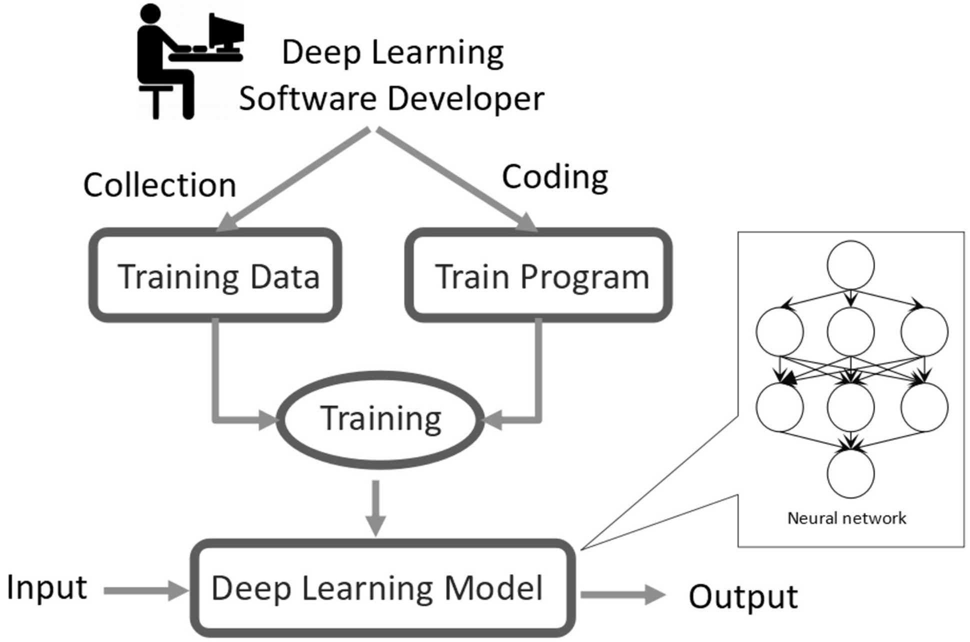
Traditional Software

Decision Logic:
- Form of code

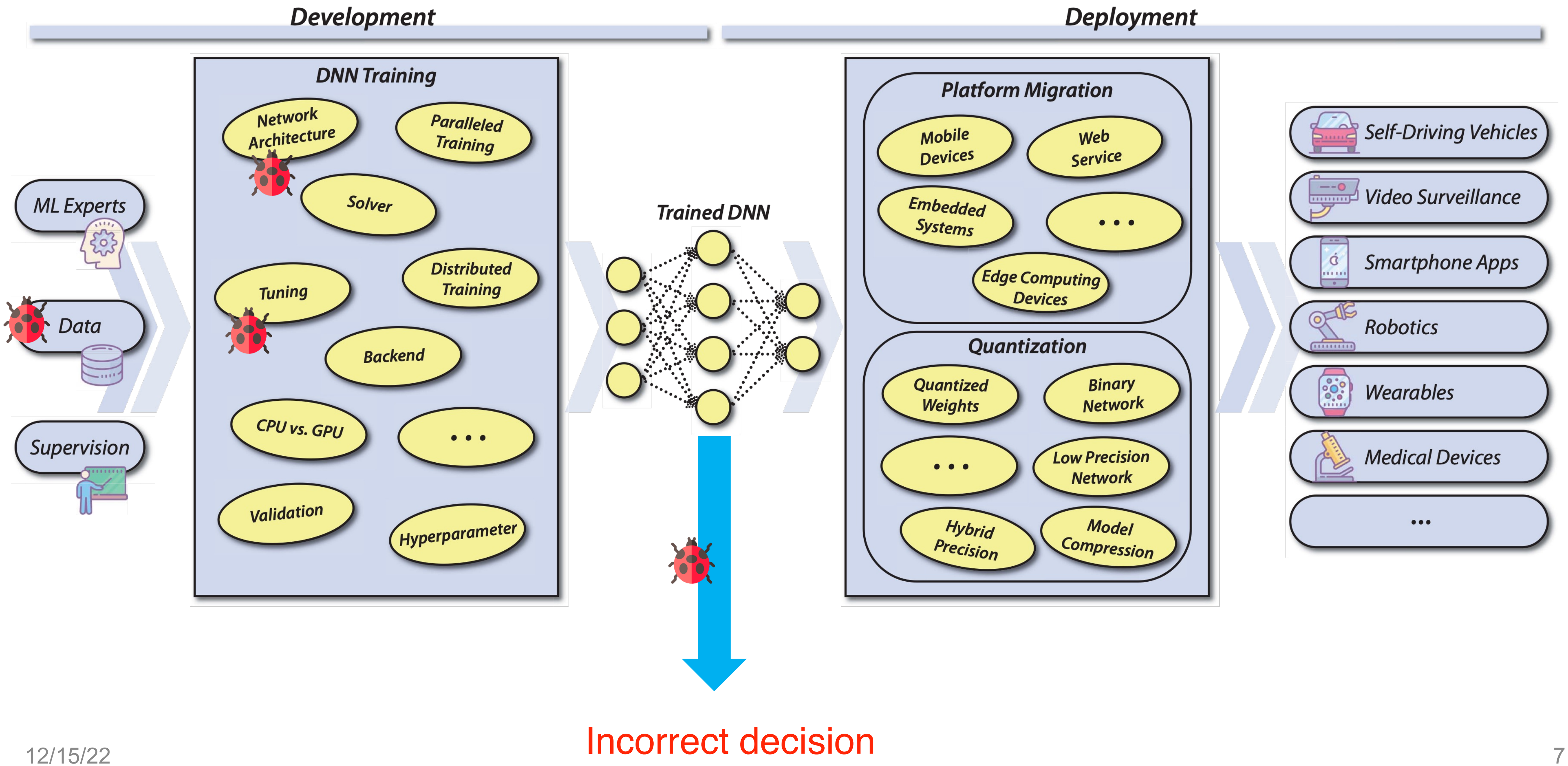


Deep Learning System

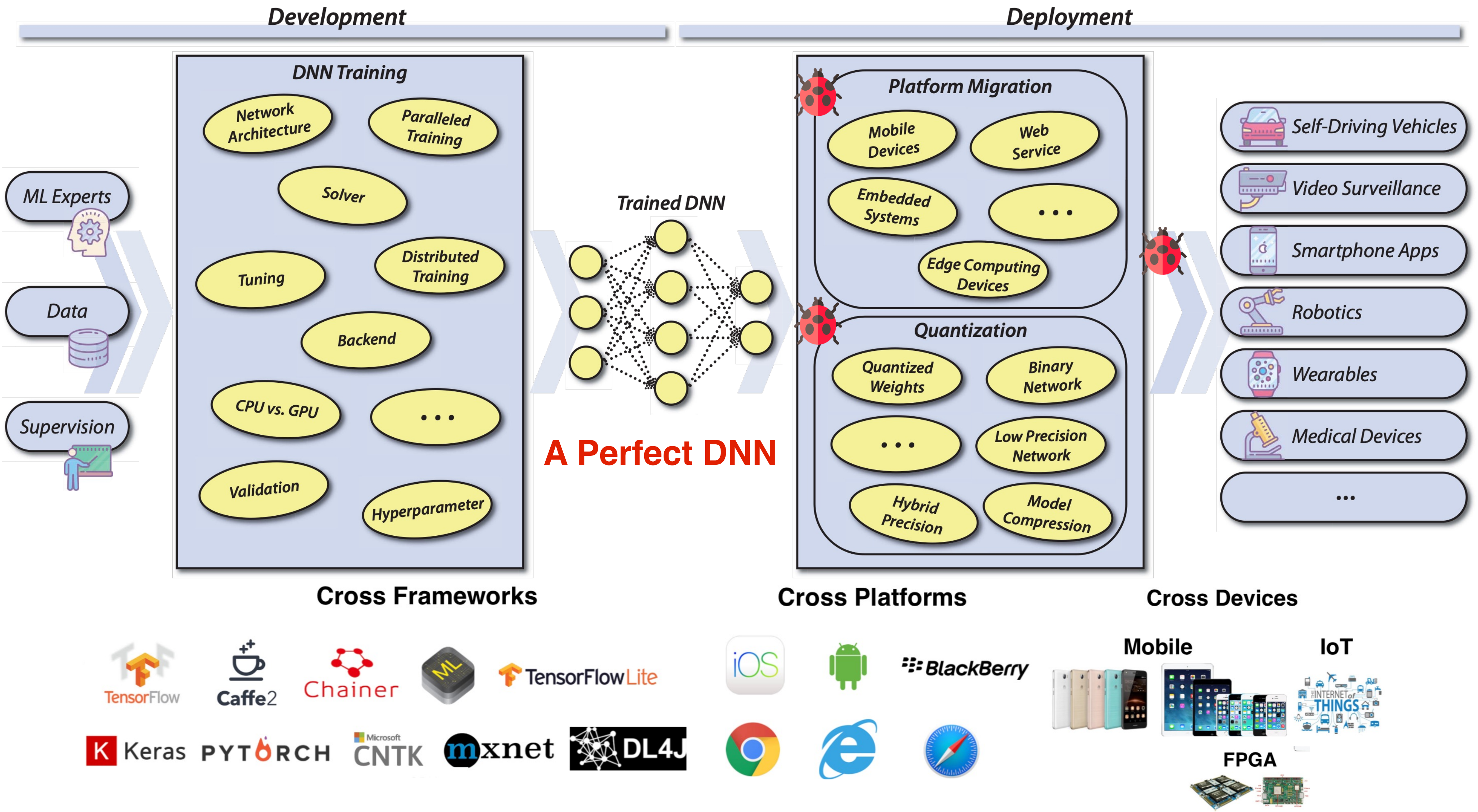
Decision Logic:
- The structure of DNN
- The connection weights



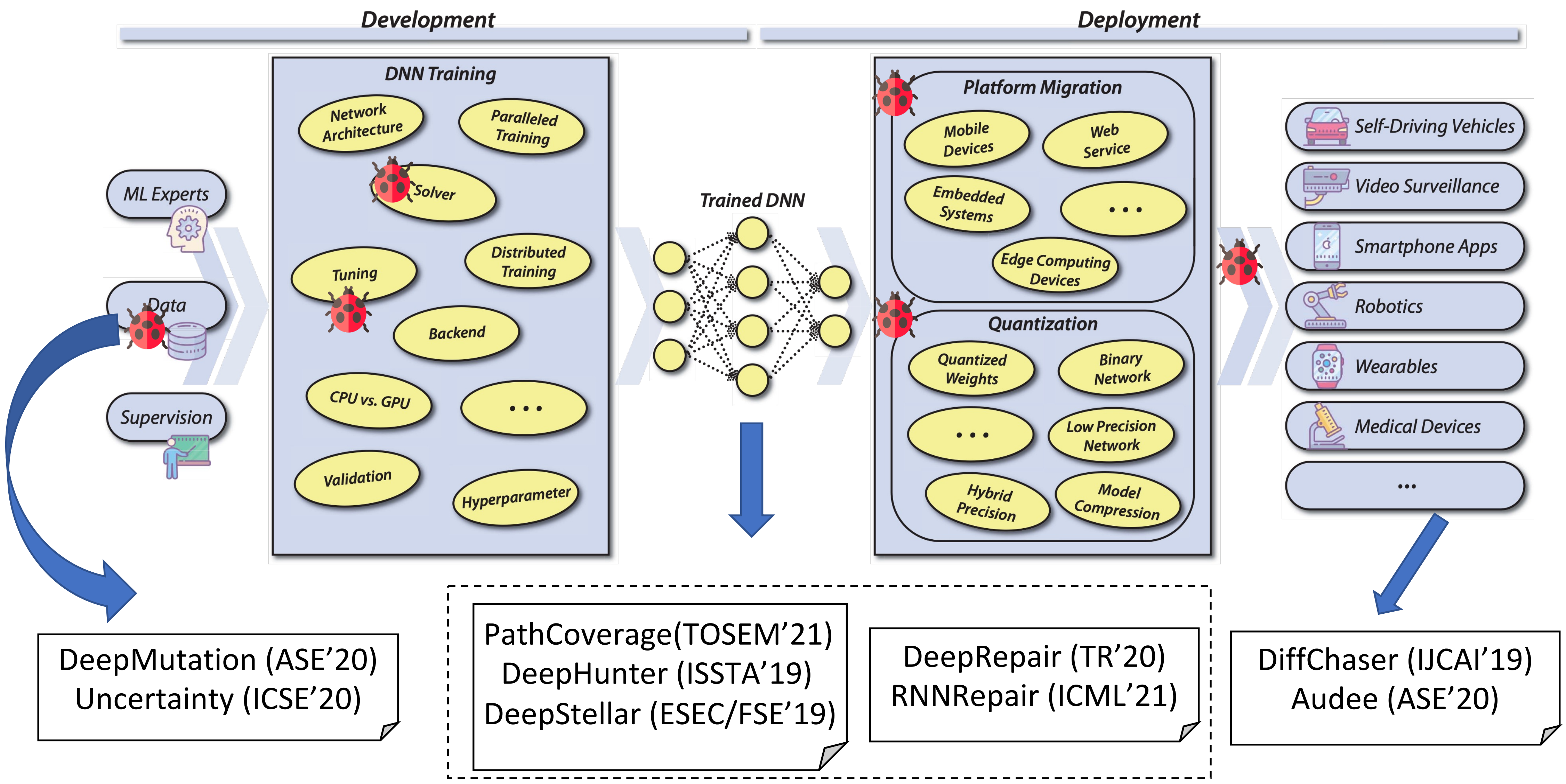
Typical DL Development and Deployment Lifecycle



Even more Challenging Issues during Deployment



Full Stack Automated Testing and Analysis Solutions



Neuron Path Coverage via Characterizing Decision Logic of Deep Neural Networks (TOSEM 2021)

Traditional Code Coverage Criteria

- Line Coverage
- Branch Coverage
- Function Coverage
- Path Coverage

Structural DNN Coverage Criteria

- Neuron Coverage (SOSP'17)
- DeepGauge (ASE'18) - KMNC, NBC, SNAC, TKNC
- DeepCT (SANER'18)
- …

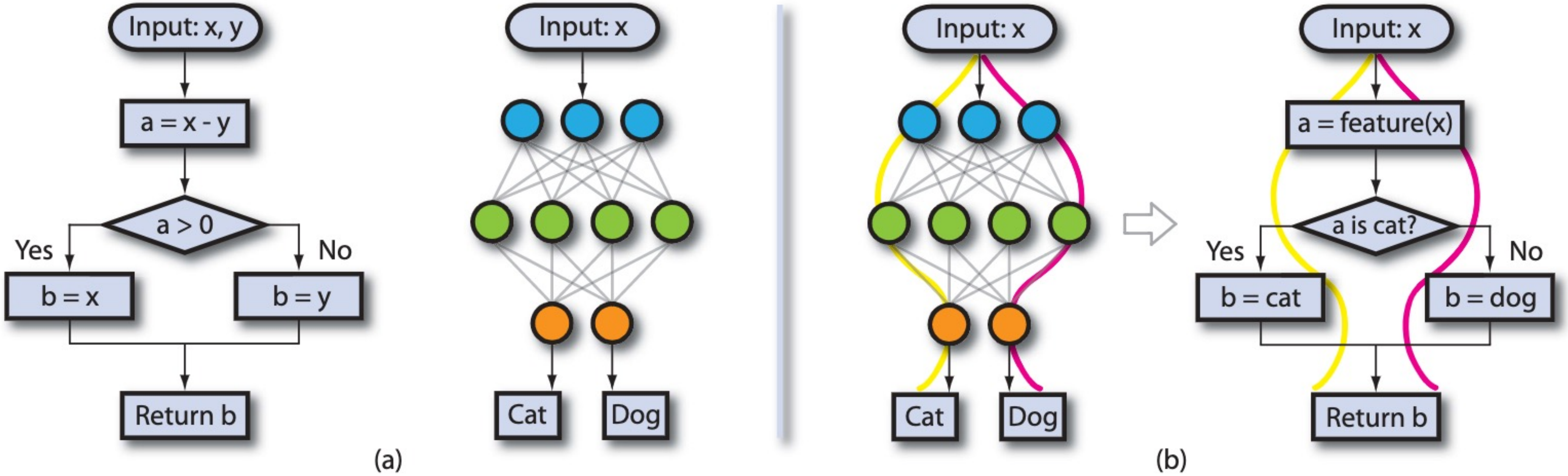
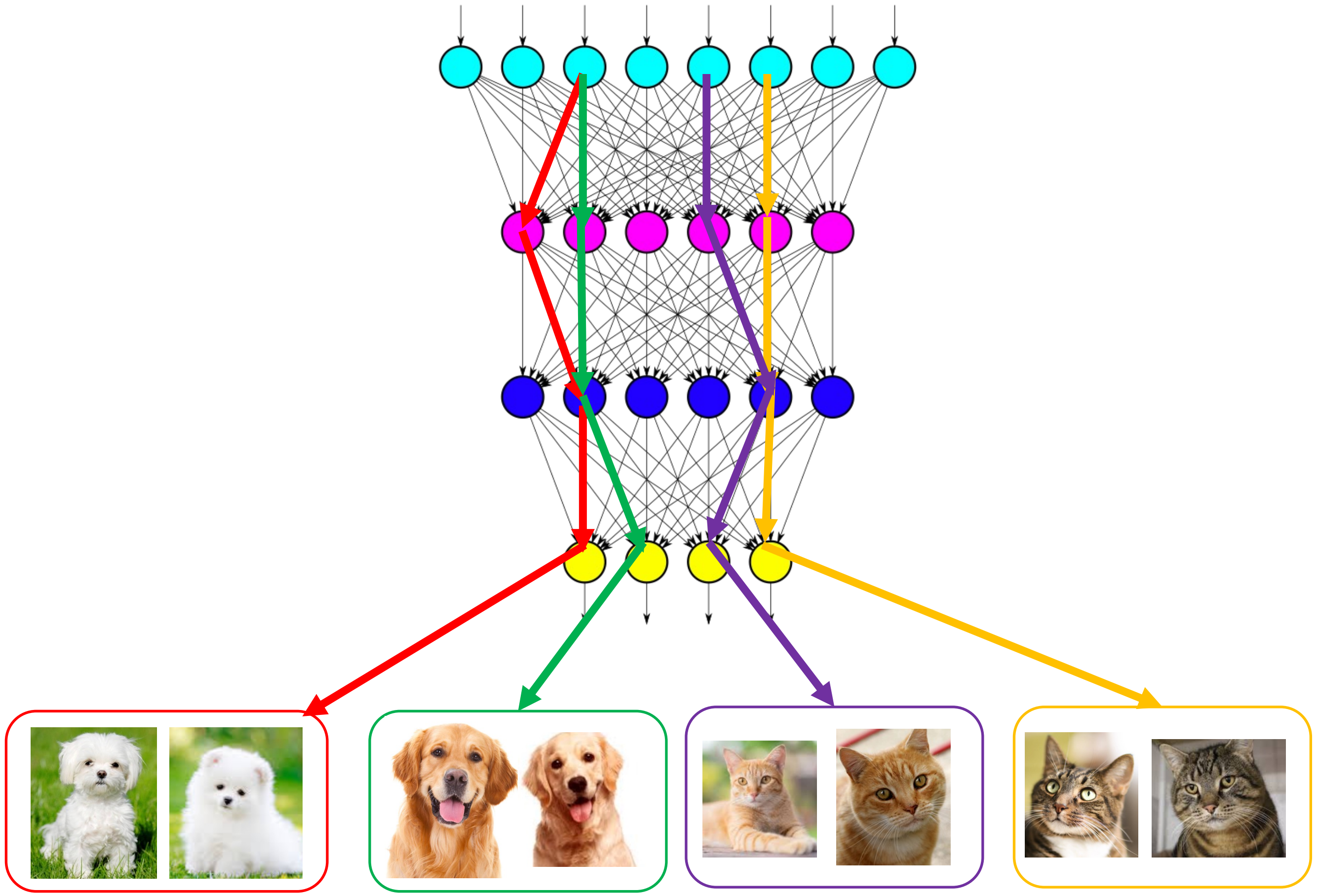


Fig. 1. (a) Difference between traditional software and DL software, (b) Paths in the DL software.

Semantics of DNN Decision – Critical Paths



Overview

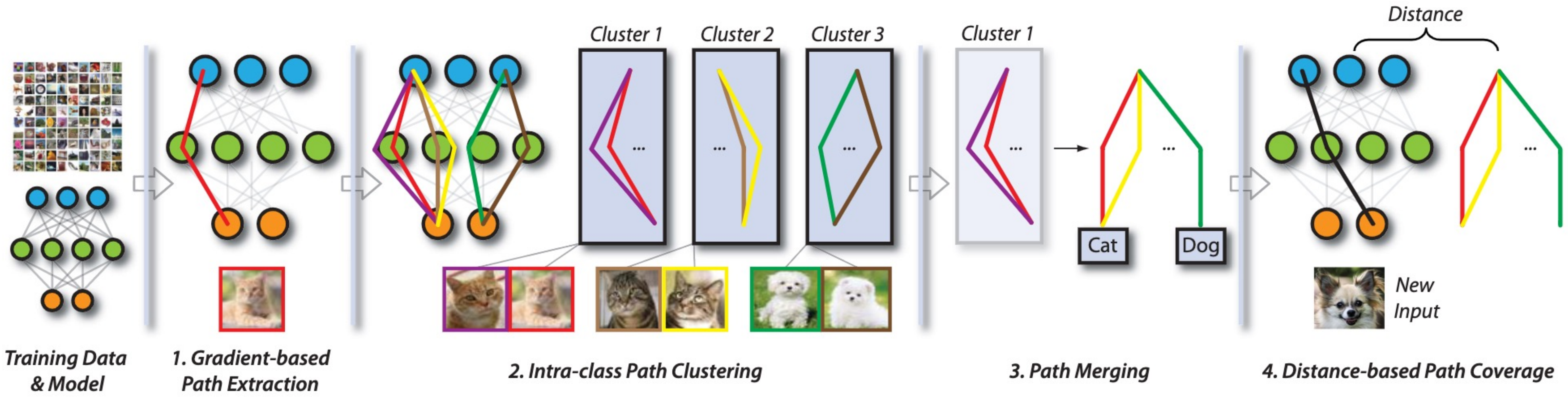


Fig. 3. Overview of this work.

- Structure-based Neuron Path Coverage (SNPC)

Given a test suite X, we define the Structure-based Neuron Path Coverage as follows:

$$SNPC(X) = \frac{|\{b^l_{x,\hat{p}} | \forall x \in X, \forall \hat{p} \in G_{f(x)}, \forall l \in f\}|}{n \cdot k \cdot |\hat{p}| \cdot m} \tag{2}$$

where $\hat{p}$ is the corresponding abstract CDP, $|\hat{p}|$ is the total number of layers, n is the number of the total classes, k is the number of clusters in the class $f(x)$ and m is the number of buckets.

$$b^l_{x,\hat{p}} = b_i \text{ if } J_{p^l_x,\hat{p}^l} \in (\frac{i-1}{m}, \frac{i}{m}] \quad J_{p^l_x,\hat{p}^l} = \frac{s^x_l \cap \hat{s}_l}{s^x_l \cup \hat{s}_l}$$

- Activation-based Neuron Path Coverage (ANPC)

Given a test suite X, we define the Activation-based Neuron Path Coverage (ANPC) as:

$$ANPC(X) = \frac{|\{d^l_{x,\hat{p}} | \forall x \in X, \forall \hat{p} \in G_{f(x)}, \forall l \in f\}|}{n \cdot k \cdot |\hat{p}| \cdot m} \tag{3}$$

$$d^l_{x,\hat{p}} = b_i \text{ if } D^l_{x,x'} \in (U \cdot \frac{i-1}{m}, U \cdot \frac{i}{m}] \quad D^l_{x,x'} = ||A(x, \hat{p}^l) - A(x', \hat{p}^l)||$$

x’ is the training sample that is the closest one to x in the cluster j

Evaluation 1 – Effectiveness of Path Abstraction

Table 4. The average width and inconsistency rate (%) after masking neurons in the abstract CDP and NCDP

(k, β)	SADL-1			SADL-2			VGG16(CIFAR-10)			AlexNet			VGG16(ImageNet)		
	Wid.	Inc.C	Inc.NC	Wid.	Inc.C	Inc.NC	Wid.	Inc.C	Inc.NC	Wid.	Inc.C	Inc.NC	Wid.	Inc.C	Inc.NC
(1, 0.6)	16.9	93.6	2.3	13.9	99.9	2.4	15.5	99.8	4.3	17.3	99.4	4.9	41.5	99.8	1.7
(1, 0.7)	14.6	89.2	8.4	9.4	99.8	2.4	13.1	99.3	4.3	13.6	99.1	13.9	34.4	99.8	1.7
(1, 0.8)	12.5	78.8	29.9	6.4	99.2	2.4	10.8	98.8	4.3	9.8	96.3	16.3	26.4	99.8	1.7
(1, 0.9)	10.4	73.6	59.8	4.1	90.9	2.4	7.9	100	10.1	5.7	50.9	37.6	17.7	99.8	1.7
(4, 0.6)	16.7	94.5	2.4	14.6	99.9	2.0	15.8	99.9	4.3	**18.5**	**99.5**	**4.4**	54.5	100	2.9
(4, 0.7)	15.4	95	2.5	10.8	99.9	2.0	13.5	99.9	4.8	15.1	99	5.8	**49**	**100**	**1.9**
(4, 0.8)	**14.6**	**94.1**	**3.9**	7.6	99.6	2.0	11.2	99.9	4.8	11.8	96.4	16.2	40.4	100	2.67
(4, 0.9)	12.2	88.1	14.3	4.8	95.3	2.0	8.4	100	9.2	7.6	80	26.7	34.3	100	2.6
(7, 0.6)	16.9	96.3	2.6	14.9	99.9	1.5	15.9	99.8	2.8	18.4	99.4	5.6	64.2	100	4.3
(7, 0.7)	15.9	96.1	3.2	11.2	99.8	1.5	13.7	99.9	3.5	15.2	98.9	6	59.5	100	2.9
(7, 0.8)	14.6	92.5	2.9	7.9	99.7	1.5	11.4	99.9	4.4	12.1	97.6	9.7	51.8	100	3.6
(7, 0.9)	12.5	88.7	9.5	**5.4**	**98.2**	**1.6**	**8.7**	**100**	**4.9**	8	90.5	19	46.7	100	3.1
(10, 0.6)	16.9	95.3	1.7	15.2	99.9	1.1	15.9	99.9	2.0	18.5	99.4	5.2	73.2	100	4.3
(10, 0.7)	16.1	95.1	3.1	11.5	99.8	1.2	13.7	99.9	2.7	15.4	98.9	6.3	70.5	100	4.3
(10, 0.8)	14.6	93.0	4.3	8.3	99.8	1.3	11.5	99.9	3.4	12.4	98.4	8.5	58.3	100	3.9
(10, 0.9)	13.5	89.2	8.8	5.4	98.8	1.3	8.7	99.9	8.8	8.6	92.9	17.7	54.3	100	2.7

Evaluation 1 – Examples

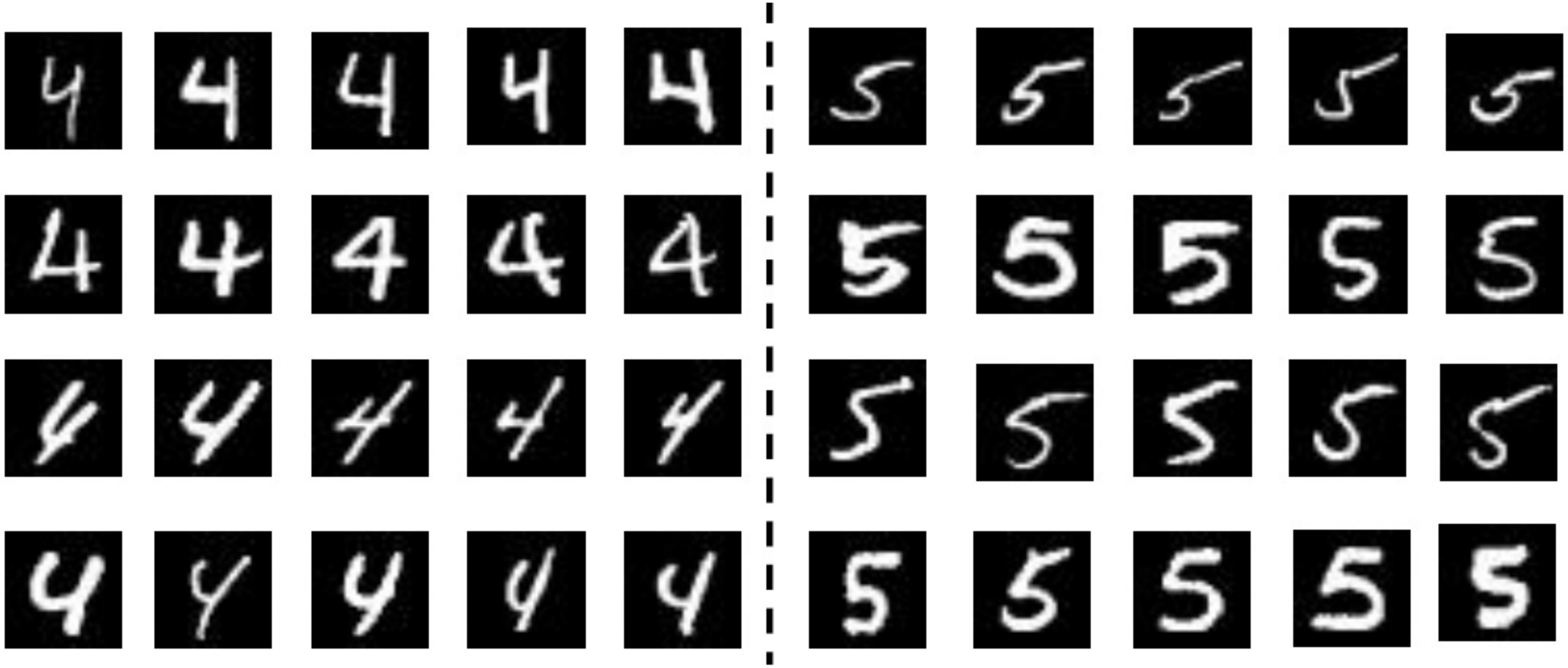


Fig. 4. Inputs sampled from clusters in the class 4 and 5

Evaluation 2 – Sensitivity with Defect Detection

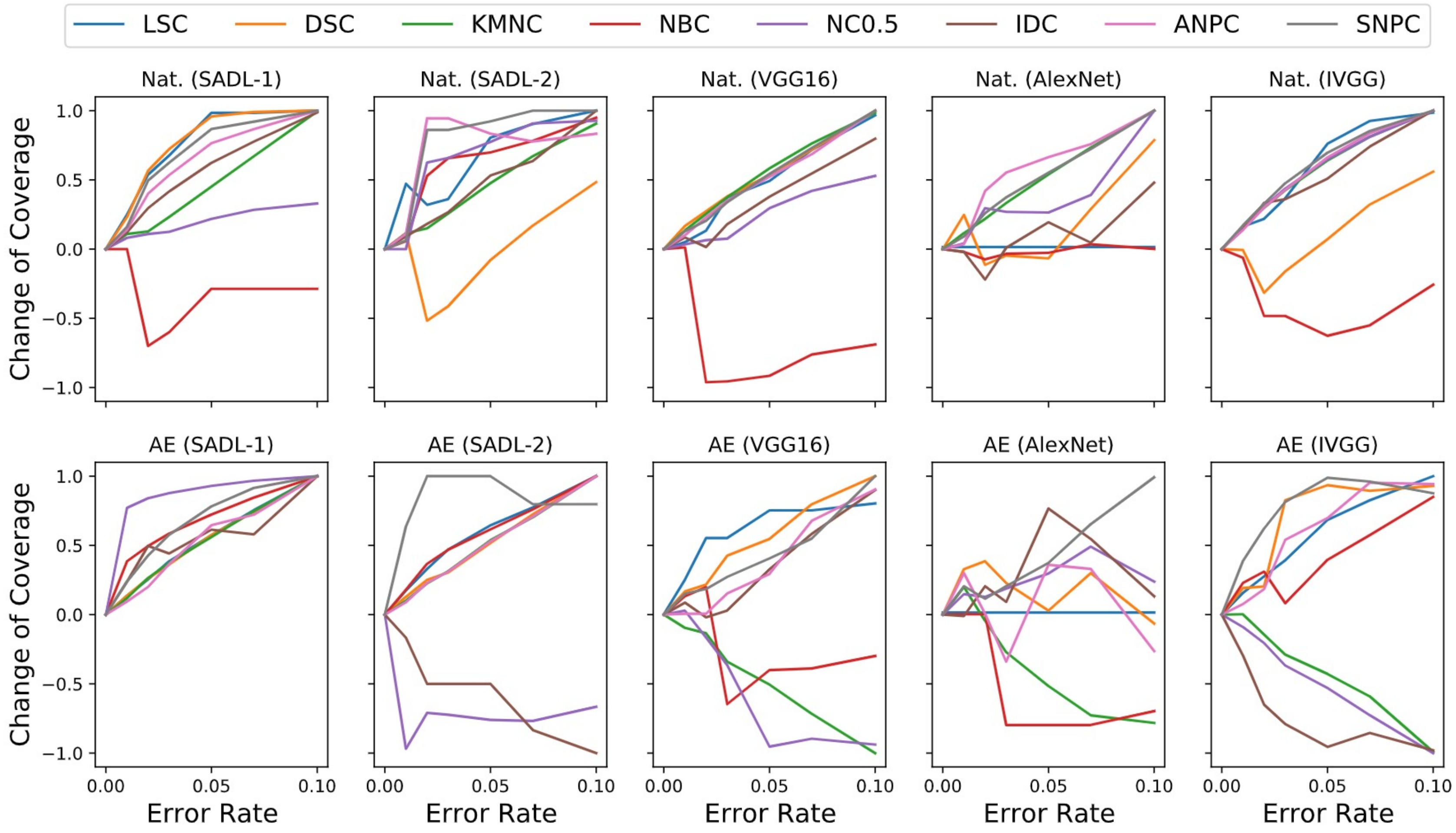


Fig. 5. Coverage change on test suites including different number of errors.

Evaluation 3 – Correlation with Output Impartiality

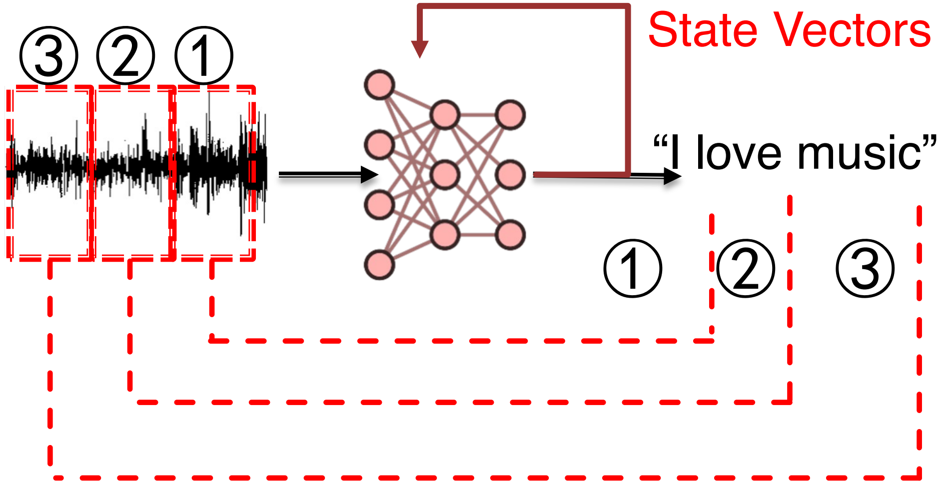
Table 6. Correlation between coverage criteria and output impartiality

		KMNC	NBC	NC(0.0)	NC(0.2)	NC(0.5)	NC(0.75)	LSA	DSA	IDC	ANPC	SNPC
size=100	SADL-1	-0.262	-0.262	-0.300	0.099	0.315	0.001	0.000	-0.638	-0.360	-0.244	**0.723**
	SADL-2	-0.029	-0.029	**0.553**	-0.397	-0.065	-0.111	0.000	0.181	**0.587**	**0.482**	**0.789**
	VGG16	-0.348	-0.348	0.037	-0.031	-0.210	-0.273	0.000	-0.145	0.052	0.012	-0.055
	AlexNet	-0.457	**0.593**	0.000	0.000	0.000	0.000	0.000	-0.320	**0.912**	**0.961**	**0.989**
	Avg.	-0.274	-0.011	0.072	-0.082	0.010	-0.096	0.000	-0.230	**0.298**	**0.601**	**0.612**
size=500	SADL-1	**0.639**	-0.116	0.000	0.000	0.000	0.000	0.000	-0.639	-0.707	**0.340**	**0.870**
	SADL-2	**0.543**	0.041	0.000	0.000	0.000	0.000	0.000	0.330	**0.473**	**0.584**	**0.817**
	VGG16	**0.414**	**0.589**	0.000	0.000	0.000	0.000	0.000	-0.255	**0.210**	**0.141**	**0.821**
	AlexNet	**0.539**	**0.586**	0.000	0.000	0.000	0.000	0.000	-0.170	**0.931**	**0.908**	**0.974**
	Avg.	**0.534**	0.275	0.000	0.000	0.000	0.000	0.000	-0.184	**0.227**	**0.893**	**0.871**

*Harel-Canada, Fabrice, Lingxiao Wang, Muhammad Ali Gulzar, Quanquan Gu, and Miryung Kim. "Is neuron coverage a meaningful measure for testing deep neural networks?." ESEC/FSE pp. 851-862. 2020.

Model-Based Quantitative Analysis of Stateful Deep Learning Systems

(ESEC/FSE’19)



Internal state transition:

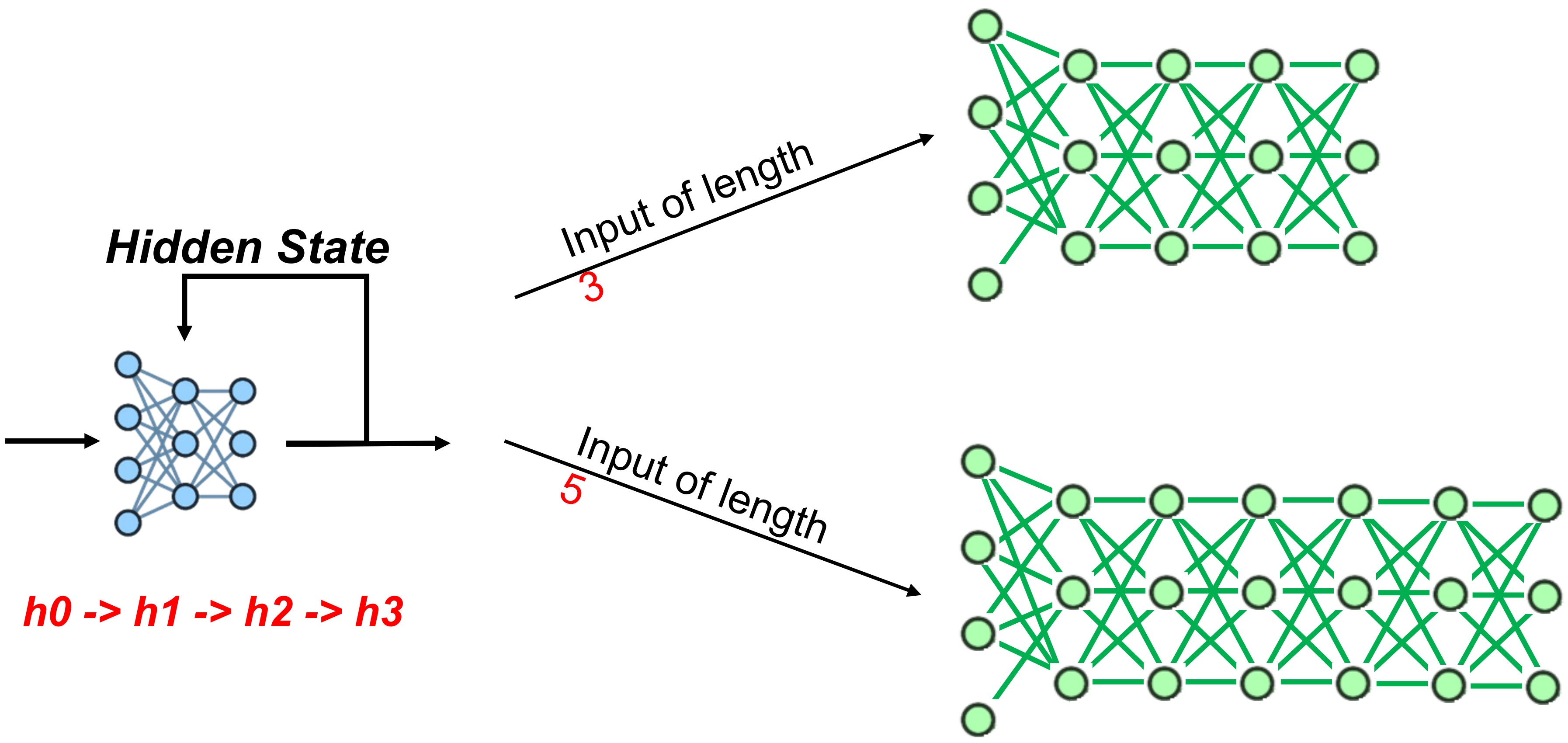
h0 -> h1 -> h2 -> h3

Review (X)	Rating (Y)
"This movie is fantastic! I really like it because it is so good!"	★★★★☆
"Not to my taste, will skip and watch another movie"	★★☆☆☆
"This movie really sucks! Can I get my money back please?"	★☆☆☆☆

Sentiment Analysis

Video Activity Recognition

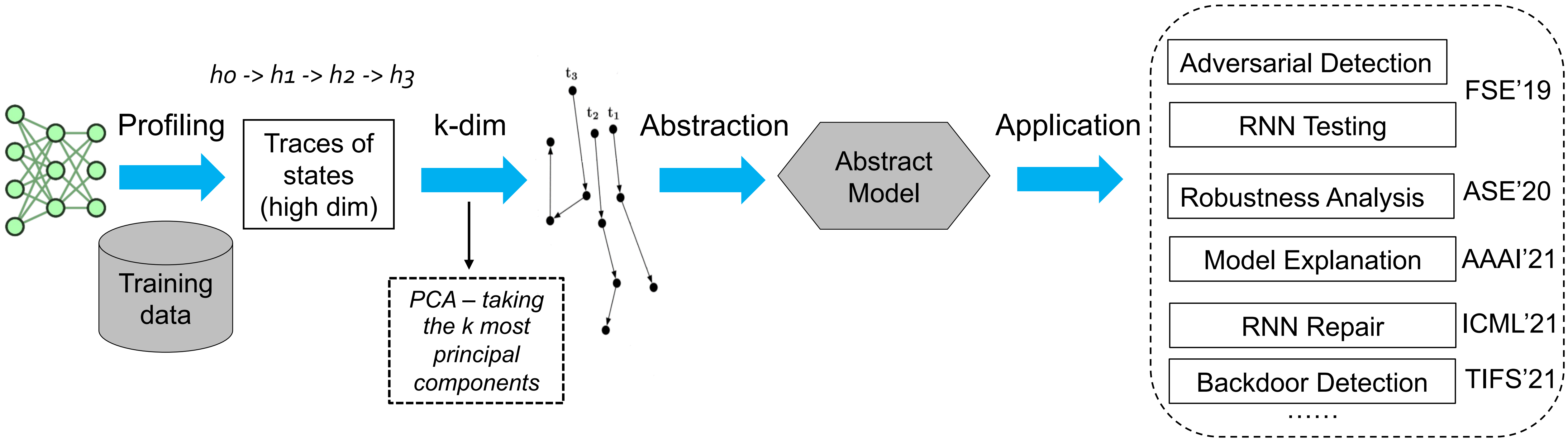
RNN is different with FNN



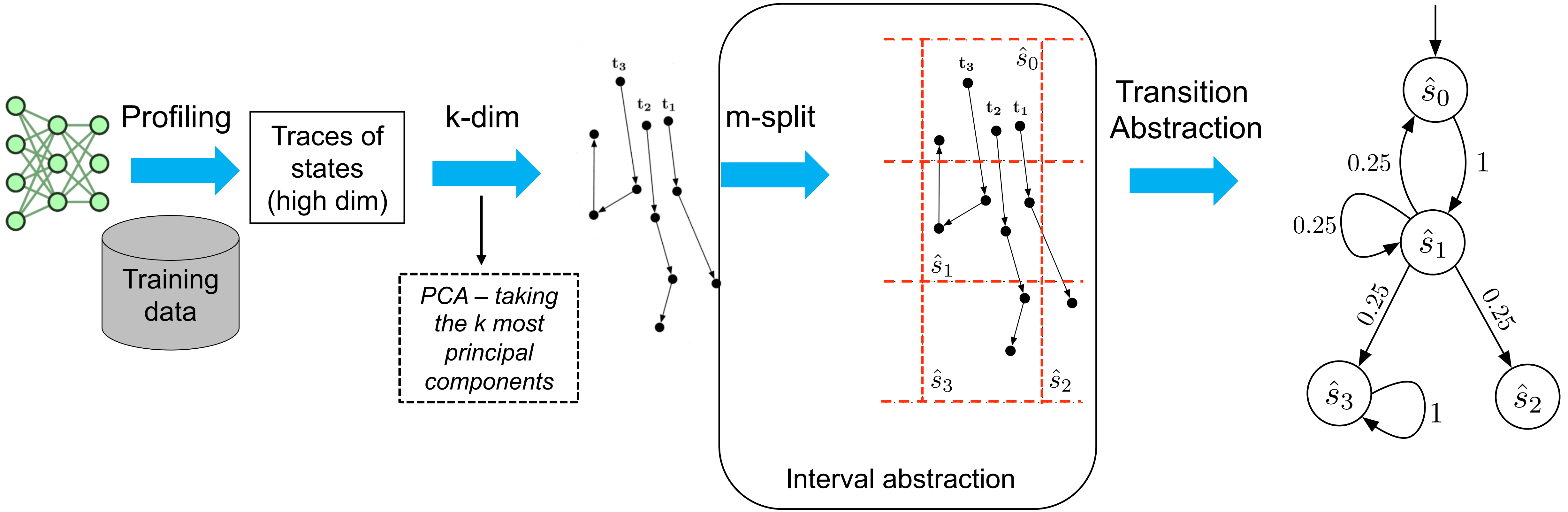
Build Abstract Model for RNN

Key insight: the logic of DNN is determined by the training data.

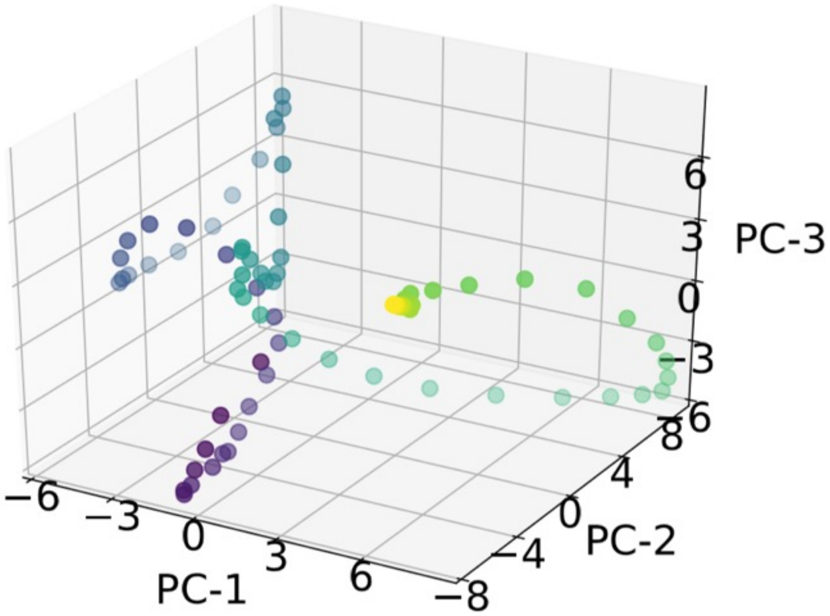
Build an abstract model by observing behaviors of RNN on training data.



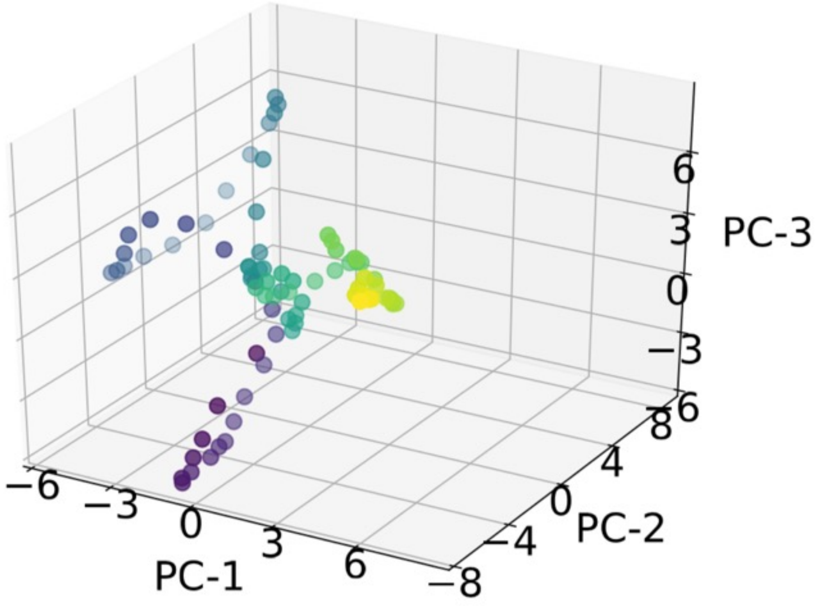
Abstracted as Discrete-Time Markov Chain (ESEC/FSE'19)



Two examples about traces in the abstract model



“This book is about science.”



“This book is about literature.”

Similarity metrics and coverage criteria for quantitative analysis

For any two samples

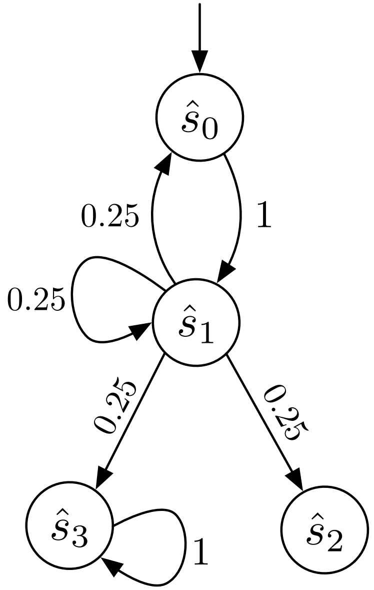
Similarity Metrics

$$\text{STSim}_M(\mathbf{x},\mathbf{y}) = \frac{|\hat{S}_x \cap \hat{S}_y|}{|\hat{S}_x \cup \hat{S}_y|}$$

State-based Trace Similarity Metrics (*STSim*)

$$\text{TTSim}_M(\mathbf{x},\mathbf{y}) = \frac{|\hat{\delta}_x \cap \hat{\delta}_y|}{|\hat{\delta}_x \cup \hat{\delta}_y|}$$

Transition-based Trace Similarity Metrics (*TTSim*)



Coverage Criteria

For Sample Set

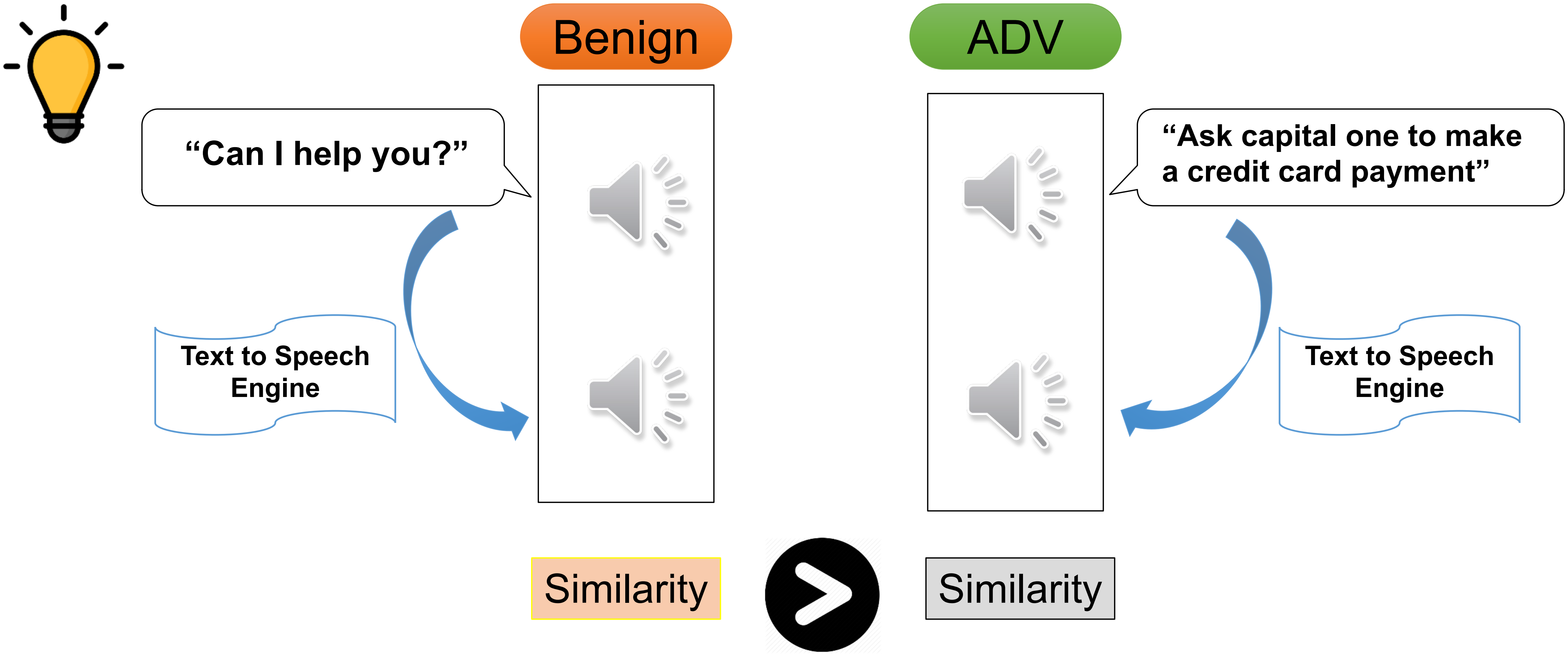
Basic State Coverage (*BSCov*)

Weighted State Coverage(*WSCov*)

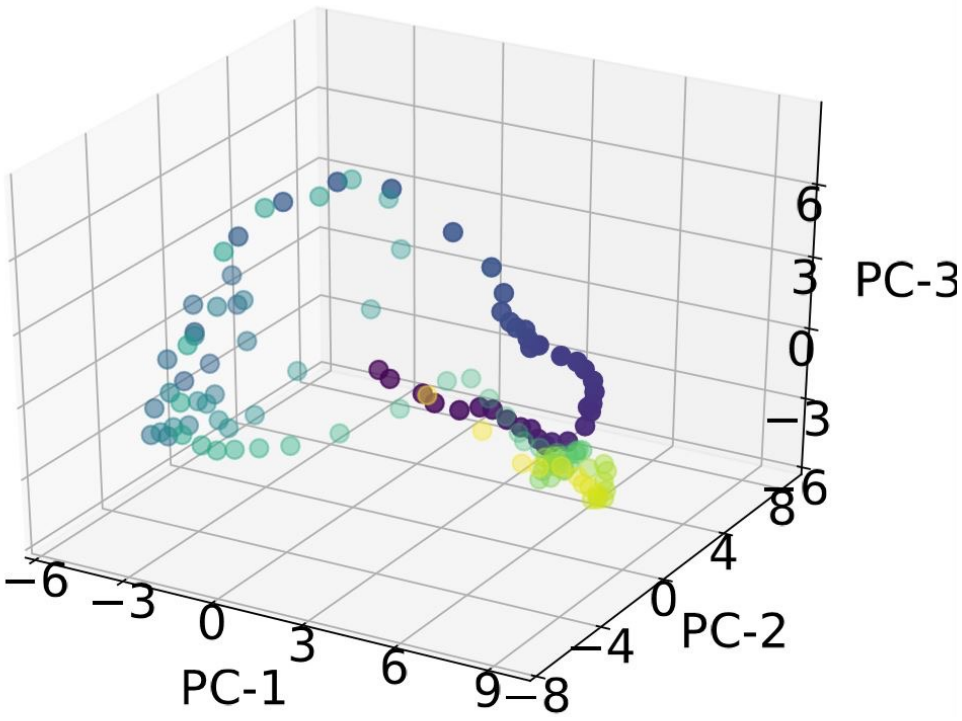
n-step State Boundary Coverage (*n-SBCov*)

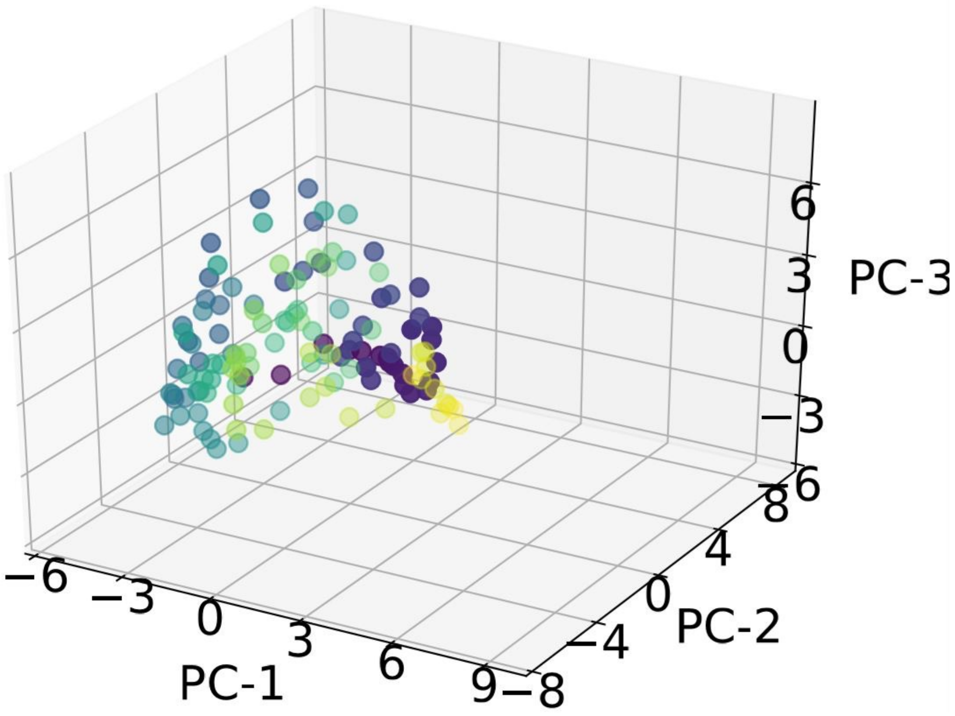
Basic Transition Coverage (*BTCov*)

Weighted Transition Coverage (*WTCov*)

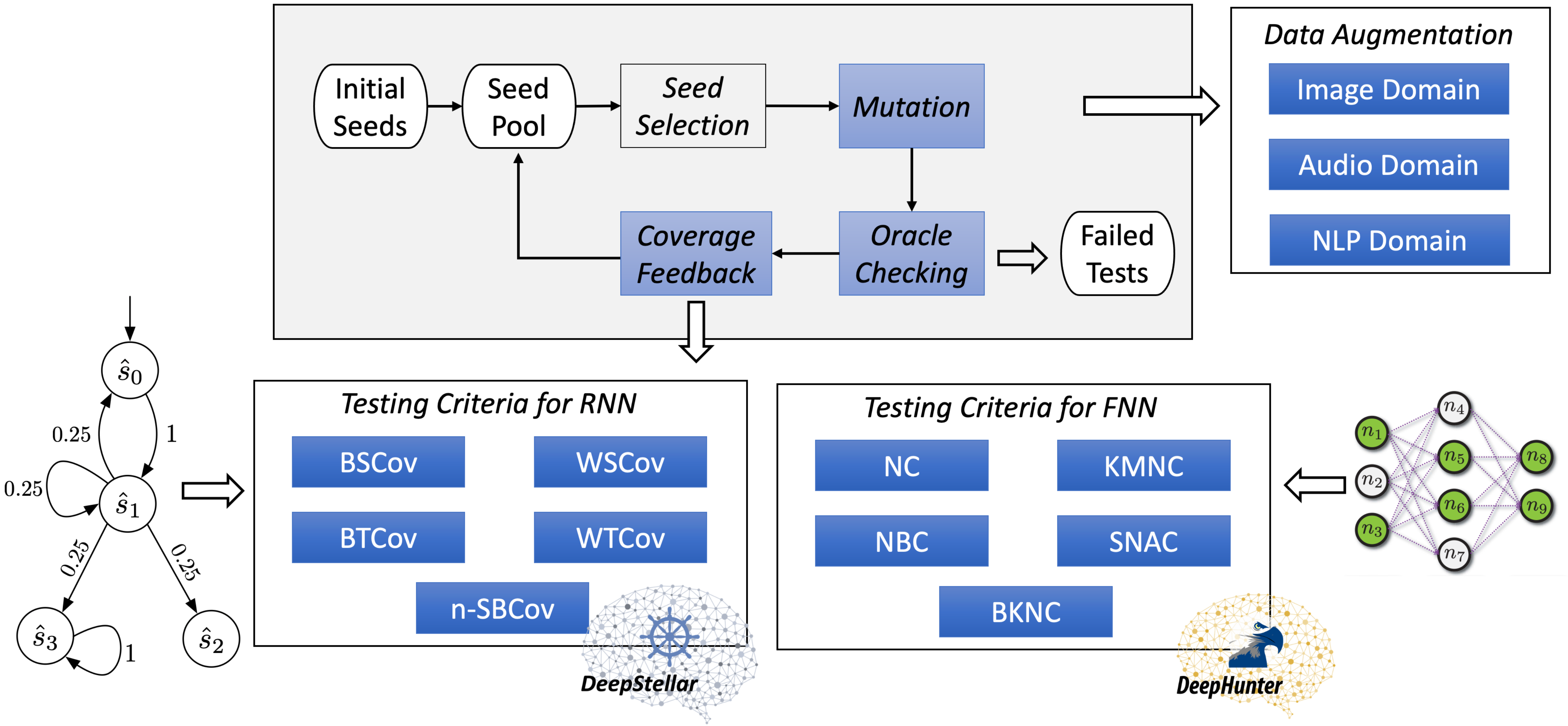

Benign
ADV
“Can I help you?”
“Ask capital one to make a credit card payment”
Text to Speech Engine
Text to Speech Engine
Similarity
Similarity

Traces of the example audio



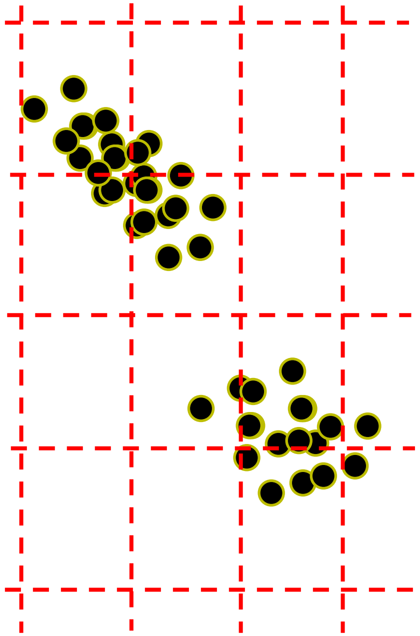


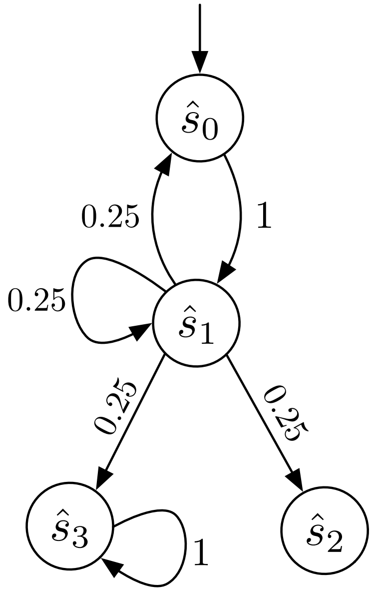
Coverage-Guided Testing



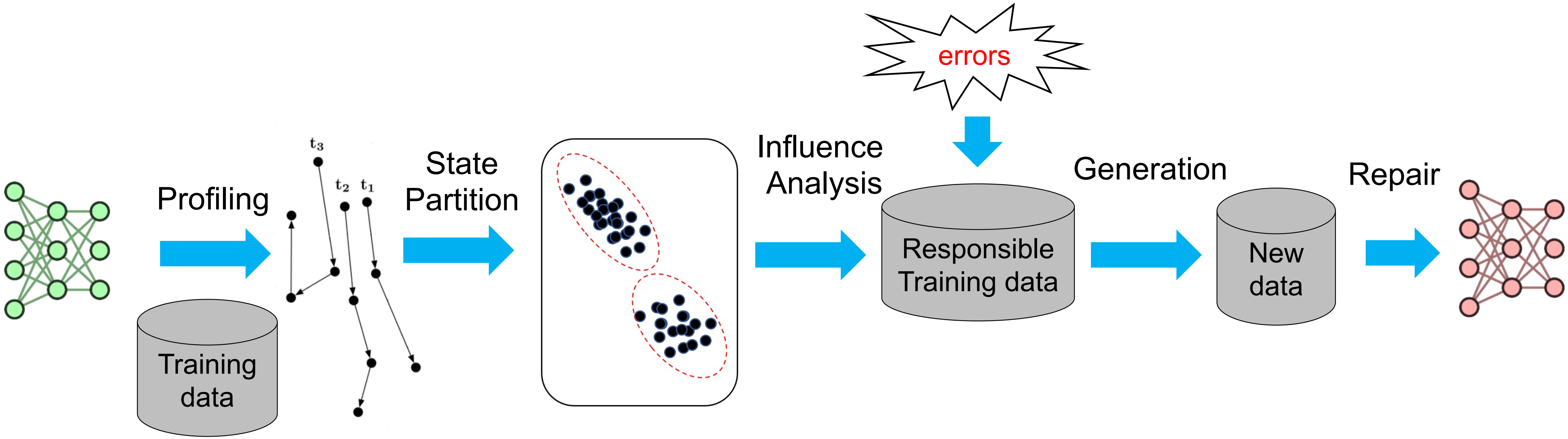
RNNRepair: Automatic RNN Repair via Model-based Analysis (ICML'21)

1. Interval abstraction is not precise
2. Semantics of the abstract model?



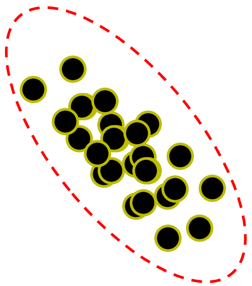


Repair via Model-based Influence Analysis



Semantics of Abstract State

Suppose RNN performs a classification task: ***f(h) = l,*** where *l* is in *{0,…,n-1}*

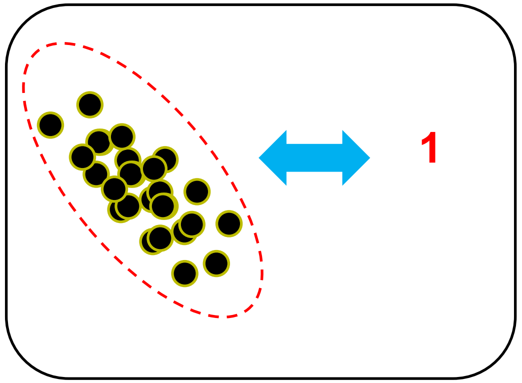


10,000

f

label	#state	confidence
0	100	**0.01**
1	9,500	**0.95**
…	…	**…**
n-1	200	**0.02**

semantics



When falling into this abstract state, it is more likely (0.95 confidence) to be predicted as 1

A stronger relation between the abstract state and a class

Example on MNIST

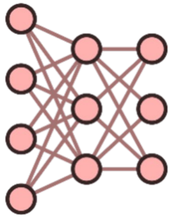
Time: ... 9 ... 11 ... 13 ... 15 ... 17 ... 20 ... 28

Input:

Time:	9	11	13	15	17	20	28
Abstract States:	7: 0.35	7: 0.35	2: 0.378	3: 0.476	0: 0.645	9: 0.493	8: 0.977
	8: 0.334	8: 0.334	3: 0.176	0: 0.322	6: 0.182	8: 0.263	9: 0.019
	3: 0.227	3: 0.227	1: 0.169	6: 0.15	9: 0.069	0: 0.096	4: 0.003
	0: 0.029	0: 0.029	8: 0.151	5: 0.027	3: 0.047	4: 0.059	7: 0.001
	2: 0.027	2: 0.027	6: 0.069	1: 0.01	5: 0.026	7: 0.056	3: 0.0
	1: 0.021	1: 0.021	5: 0.044	8: 0.005	7: 0.024	6: 0.026	2: 0.0
	9: 0.007	9: 0.007	0: 0.013	9: 0.004	8: 0.007	5: 0.006	6: 0.0
	5: 0.002	5: 0.002	9: 0.0	7: 0.003	4: 0.0	3: 0.001	5: 0.0
	6: 0.002	6: 0.002	7: 0.0	2: 0.003	2: 0.0	1: 0.001	1: 0.0
	4: 0.0	4:0.0	4: 0.0	4: 0.0	1: 0.0	2: 0.0	0: 0.0
Prediction:	7: 0.21	3: 0.49	2: 0.89	3: 0.78	0: 0.26	9: 0.32	8: 0.99

Identify training samples that have the largest impact on the predication of the test sample.

$$\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$$



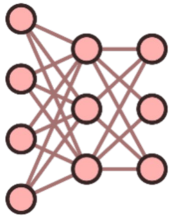
Sample-Level Influence Analysis

Segment-Level Influence Analysis

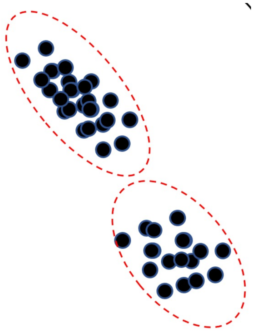
Segment-level Influence Analysis

Trace:

$$\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$$



$$(\boldsymbol{h}_0, \boldsymbol{h}_1, \ldots, \boldsymbol{h}_n)$$



$$\tau_{\boldsymbol{x}} = (q_0, \boldsymbol{x}_1, q_1, \ldots, \boldsymbol{x}_n, q_n)$$

Influence Analysis:

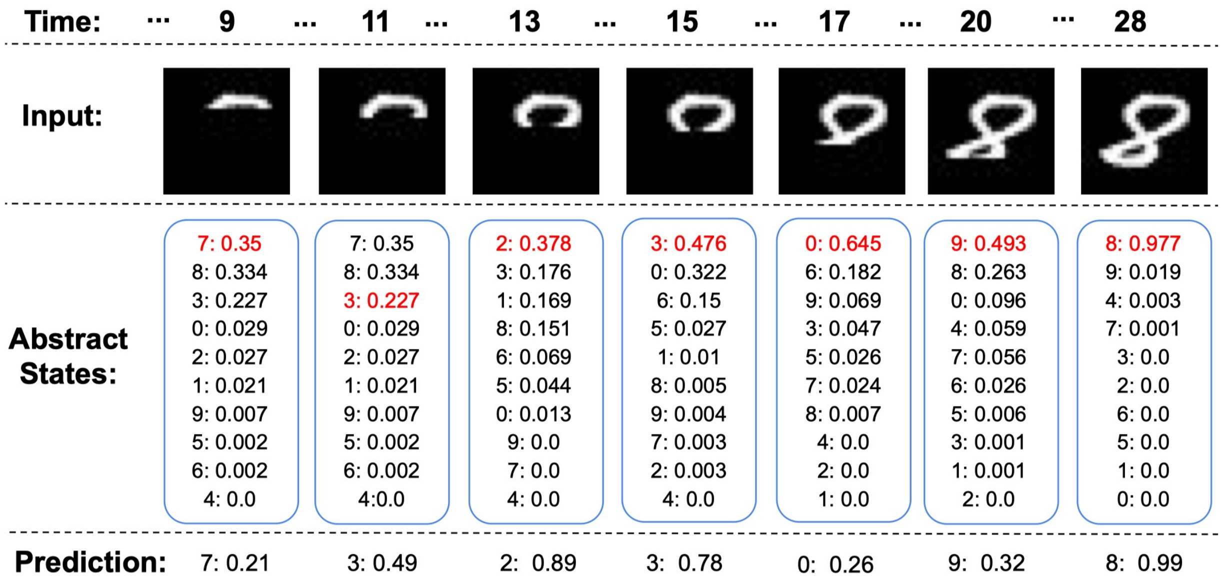
$$\forall 0 < i \leq n, \mathcal{I}(q_{i-1}, \boldsymbol{x}_i) = \mathcal{I}(q_{i-1}, \boldsymbol{x}_i) \cup \{\boldsymbol{x}\}$$

The pencil has a sharp point.
It is not polite to point at people.

Sample-level Influence Analysis

$$\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$$

$$\tau_{\boldsymbol{x}} = (q_0, \boldsymbol{x}_1, q_1, \ldots, \boldsymbol{x}_n, q_n)$$

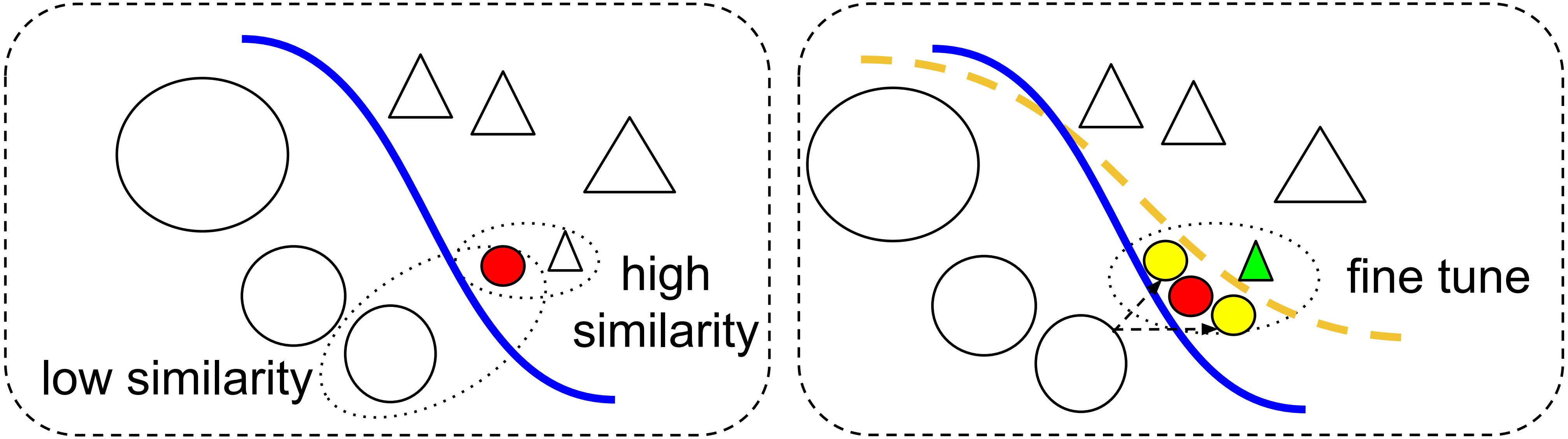


$$\mathcal{F}_{\boldsymbol{x}} = (f_0, \ldots, f_n)$$

$$where \;\; f_i = (ID(q_i), C_{q_i}, \mathcal{Y}_R^n(\boldsymbol{h}_i))$$

$$infl_{score}(\boldsymbol{x}_{\text{train}}, \boldsymbol{x}_{\text{test}}) = \text{similarity}(\mathcal{F}_{\boldsymbol{x}_{\text{train}}}, \mathcal{F}_{\boldsymbol{x}_{\text{test}}})$$

Fault Localization and Remediation (Sample-level)



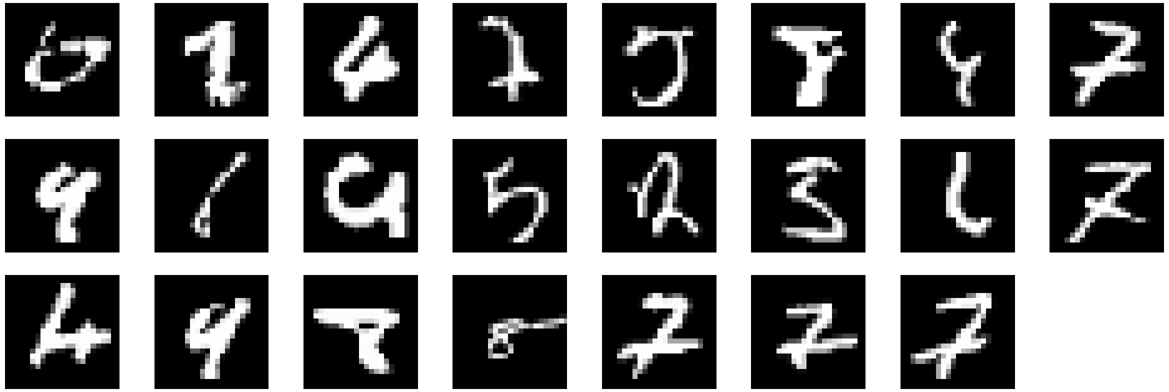
Fault segments: $$S = \{\boldsymbol{x}_i | 1 \leq i \leq n \land |\mathcal{I}(q_{i-1}, \boldsymbol{x}_i)| < \gamma\}$$

Example:

$$① \xrightarrow{Just(1{,}43)} ② \xrightarrow{noticed(1{,}11)} ③ \xrightarrow{who(1{,}19))} ④ \xrightarrow{gave(1{,}5)}$$
$$⑤ \xrightarrow{that(1{,}89)} ⑥ \xrightarrow{out(1{,}6)} ⑦ \xrightarrow{lulz(0{,}0)} ⑧ \xrightarrow{.(0{,}807)} ⑨$$

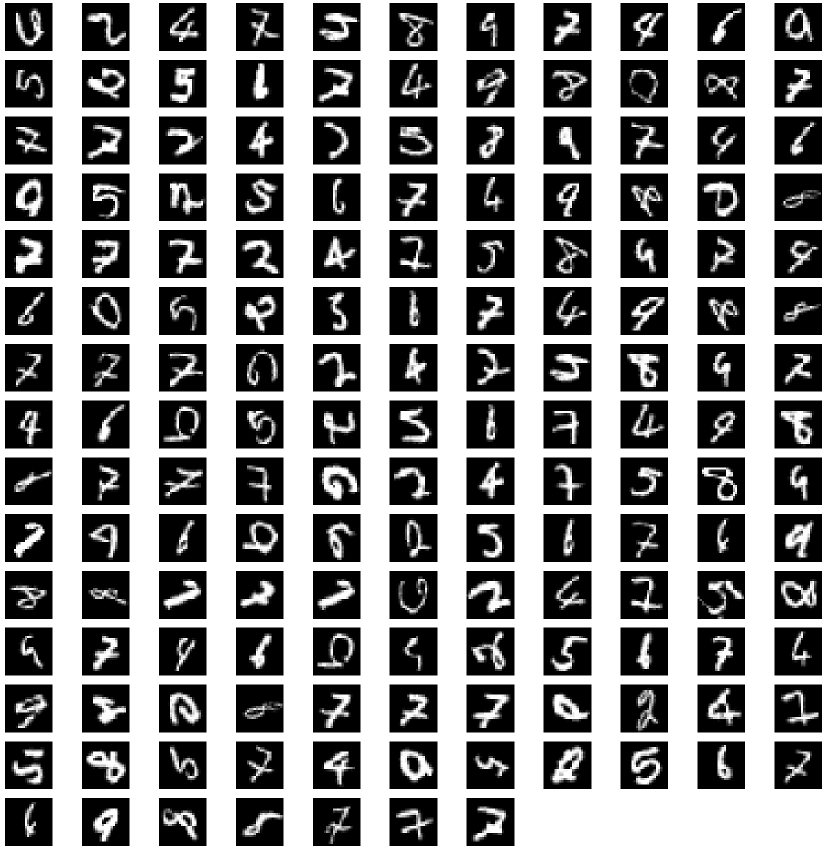
Insert ***lulz*** in the influential training samples (after q_{i-1}) that have the same prediction with the failed sample.

Evaluation 1 - Sample-level Repair on MNIST



23 failed inputs

Model	23 failed inputs	161 new inputs
Original Data	**1.3** (5.7%)	**63.9** (39.7%)
Ori + 161 New	**11.7** (50.9%)	**130.6** (81.8%)
Random	**4.3** (18.7%)	-



161 generated inputs

Evaluation 2 - Segment-level Repair on TOXIC and SST

Table 4: Results of Repairing on Toxic and SST

	Num. (m)	5	15	25	35	45
Toxic	Random	43.63%	63.18%	65.91%	66.36%	61.36%
	RNNRepair	50%	65.64%	72.73%	81.82%	81.82%
SST	Random	26.09%	21.74%	47.83%	47.83%	60.86%
	RNNRepair	30.43%	52.17%	60.87%	65.22%	65.22%

* TOXIC: Toxic Comment Classification Challenge
SST: Standard Sentiment Treebank

Toxic: 23 errors
SST: 115 errors

Summary

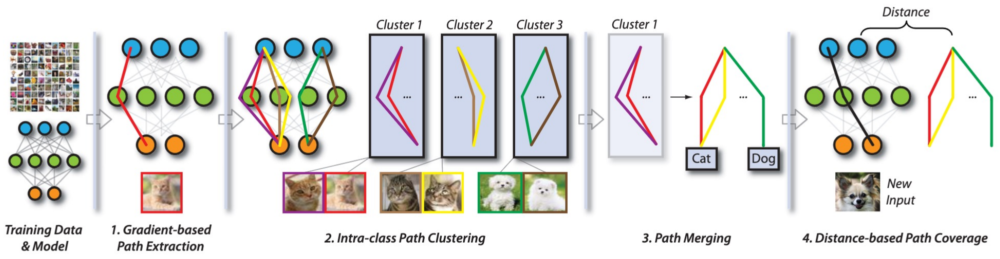
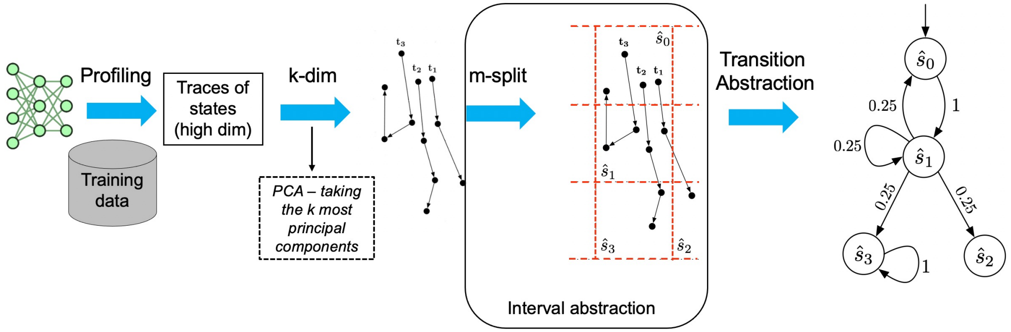
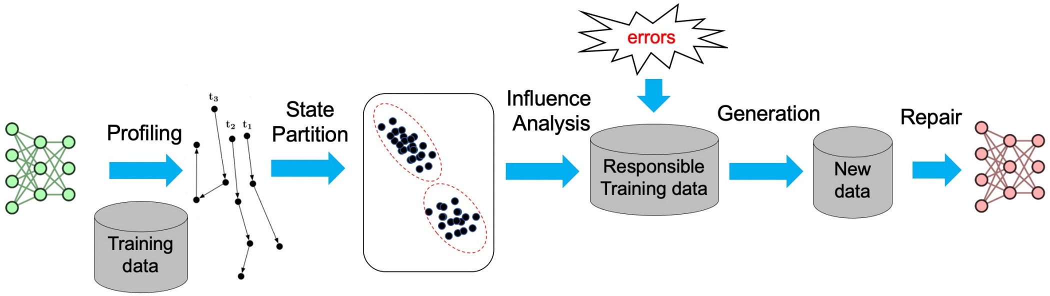


Fig. 3. Overview of this work.



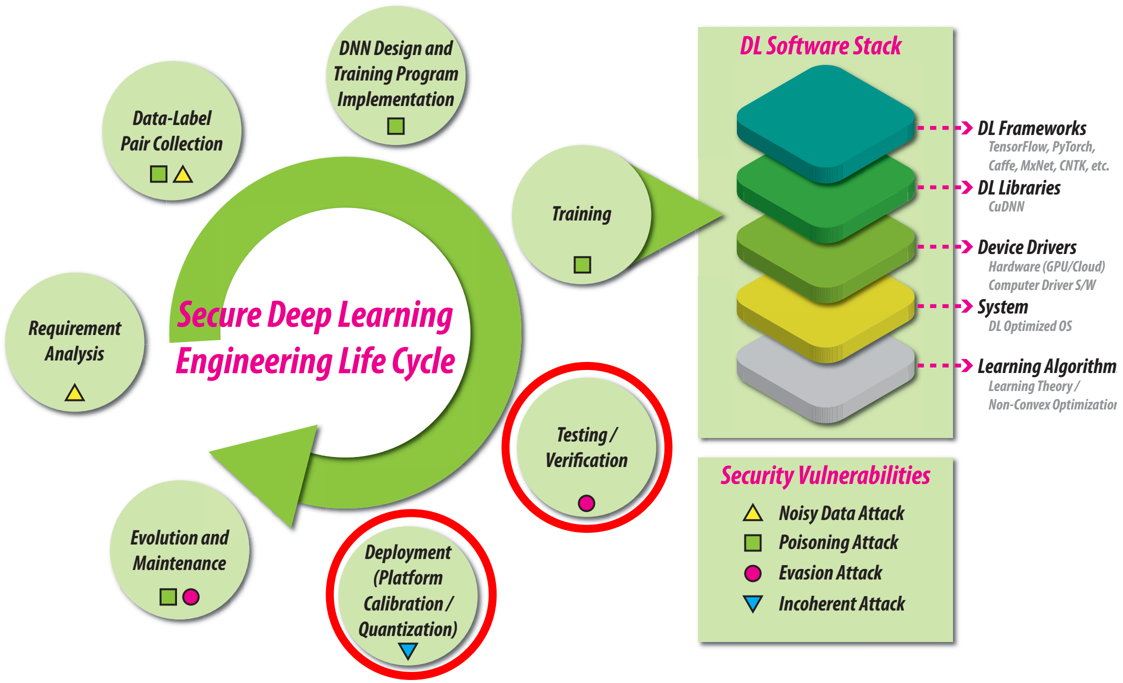


Testing Criteria

RNN Testing

RNN Repair

Interpretable Quality, Reliability and Security and Engineering Support for ML/DL Lifecycle



Life cycle → Training Process » Model » Optimization » Deployment

System Level → Model Level — Development / Deployment » Integration » System

Framework Level → Framework Level » Training » Migration » Runtime Prediction

Thanks and Questions?